

LAPORAN TUGAS 3

Race Recognition Rush



Mata kuliah	TI0263 – Kecerdasan Buatan (GRUP C)
Dosen Pengampu	Gloria Virginia, S.Kom., MAI., Ph.D
Nama Kelompok	C8
Anggota Kelompok	<ol style="list-style-type: none">1. 71230985 - Tomas Becket2. 71231002 – Philip Deric Kho3. 71231015 - Karel Marley Bala Bakior4. 71231017 - Paulus Ungirwalu5. 71231061 - Syendhi Reswara. S
Deklarasi	Dengan ini kami menyatakan bahwa tugas ini merupakan hasil karya kelompok kami, tidak ada manipulasi data serta bukan merupakan plagiasi dari karya orang lain.

Tugas3

0.1 Jelaskan 1 metode yang digunakan sebagai mesin inferensi aplikasi PAM kelompok Anda!

Aplikasi kami akan mengetahui ras dari wajah pengguna (Jawa, Cina, Papua) menggunakan face recognition berbasis knowledge based dengan aturan-aturan yang telah ditentukan. Berdasarkan data tersebut, program kami akan menentukan lokasi pada peta UKDW yang mana memiliki orang-orang dengan ras yang sama. Setelah lokasi ditentukan, sistem akan menggunakan algoritma A^* atau *Dijkstra* untuk menghitung dan menentukan rute tercepat menuju lokasi tersebut.

Contoh:

Pengguna akan diidentifikasi pada aplikasi dan ternyata memiliki ras China melalui face recognition. Sistem menggunakan metode forward chaining untuk mencocokkan hasil ini dengan aturan:

Jika ras = China, maka tujuan = Gedung Agape (jika gedung tersebut memiliki persentase ras China tertinggi)

Setelah itu, sistem mengambil lokasi pengguna saat ini (misalnya di Kantin UKDW) dan menghitung rute tercepat ke Gedung Agape menggunakan algoritma A^* atau Dijkstra. sehingga hasil akhirnya pengguna akan ditampilkan peta UKDW dengan jalur optimal menuju Gedung Agape, tempat di mana kemungkinan besar ada orang dengan ras yang sama.

0.2 Contoh kasus

There are n cities and m flight connections between them. Your task is to determine the length of the shortest route from **Syrjälä** to every city.

Input

The first input line has two integers n and m : the number of cities and flight connections. The cities are numbered $1, 2, \dots, n$, and city 1 is **Syrjälä**. After that, there are m lines describing the flight connections. Each line has three integers a , b and c : a flight begins at city a , ends at city b , and its length is c . Each flight is a one-way flight.

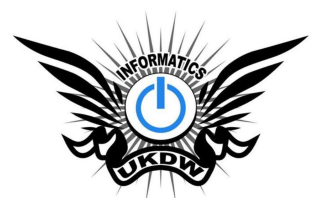
You can assume that it is possible to travel from Syrjälä to all other cities.

Output

Print n integers: the shortest route lengths from Syrjälä to cities $1, 2, \dots, n$.

Constraints

- $1 \leq n \leq 10^5$
- $1 \leq m \leq 2 \cdot 10^5$
- $1 \leq a, b \leq n$
- $1 \leq c \leq 10^9$



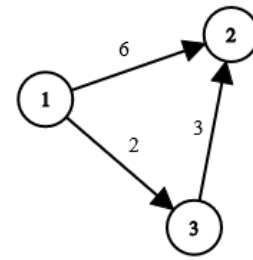
Example

Input:

```
3 4
1 2 6
1 3 2
3 2 3
```

Output:

```
0 5 2
```



```
1 #include <bits/stdc++.h>
2 using ll = long long;
3 using ull = unsigned long long;
4 using namespace std;
5
6 void solve() {
7     int n, m;
8     cin >> n >> m;
9
10    vector< vector<pair<int, int>> > neighbours(n+1);
11
12    for (int i = 1; i <= m; i++) {
13        int city;
14        pair<int, int> flight;
15        cin >> city >> flight.first >> flight.second;
16        neighbours[city].push_back(flight);
17    }
18
19    vector<ll> ans(n+1, LONG_LONG_MAX); // ans untuk answer
20    ans[1] = 0;
21
22    using apa_ini = pair<ll, int>; // saya bingung C++ template; oke sip;
23    priority_queue<apa_ini, vector<apa_ini>, greater<apa_ini>> pq;
24    pq.push({0, 1});
25
26    while (!pq.empty()) {
27        auto [distance, target] = pq.top(); pq.pop();
28        if (distance > ans[target]) continue; // menghindari pemrosesan input yang memiliki
29        jarak lebih jauh
30
31        for (int i = 0; i < neighbours[target].size(); i++) {
32            int t2 = neighbours[target][i].first;
33            ll d2 = neighbours[target][i].second + distance;
34
35            ll flag = ans[t2];
36            ans[t2] = min(ans[t2], d2);
37
38            if (ans[t2] != flag) {
39                pq.push({ans[t2], t2});
40            }
41        }
42    }
43
44    for (int i = 1; i <= n; i++) {
45        cout << ans[i] << " ";
46    } cout << '\n';
47 }
```